BAB III ANALISA DAN PERANCANGAN

Pada bab ini berisi tentang analisa dan perancangan sistem meliputi pembangunan model berdasarkan data yang sudah melalui proses tagging yang nanti akan di aplikasikan pada Stanford NER dan proses klasifikasi Analisa sentiment.

3.1 Data

Zomato.com adalah sebuah panel website atau situs pencarian restoran yang menyediakan informasi berbagai tempat makan maupun cafe. Data pada penelitian ini bersumber dari zomato.com dimana data di ambil dari 4 restoran yang berbeda.

3.1.1 Pengumpulan Data

Pengumpulan data penelitian kali ini di lakukan secara manual pada website zomato.com, data di ambil dari 4 restoran yang berbeda, 4 restoran tersebut terdiri dari:

1) Nasi goreng apjay, Jakarta, dengan 312 ulasan
2) Bakmi karet foak, Jakarta, dengan 130 ulasan
3) Bebek goreng khas Surabaya, jakarta, dengan 53 ulasan
4) Warung bu Kris, Surabaya, dengan 54 ulasan

Sehingga data kesuluruhan pada peneltian ini berjumlah 549 data, yang berasal dari 4 restoran yang sudah di paparkan pada bagian atas, dan data tersebut di simpan dengan format txt.

3.1.2 Analisa Data

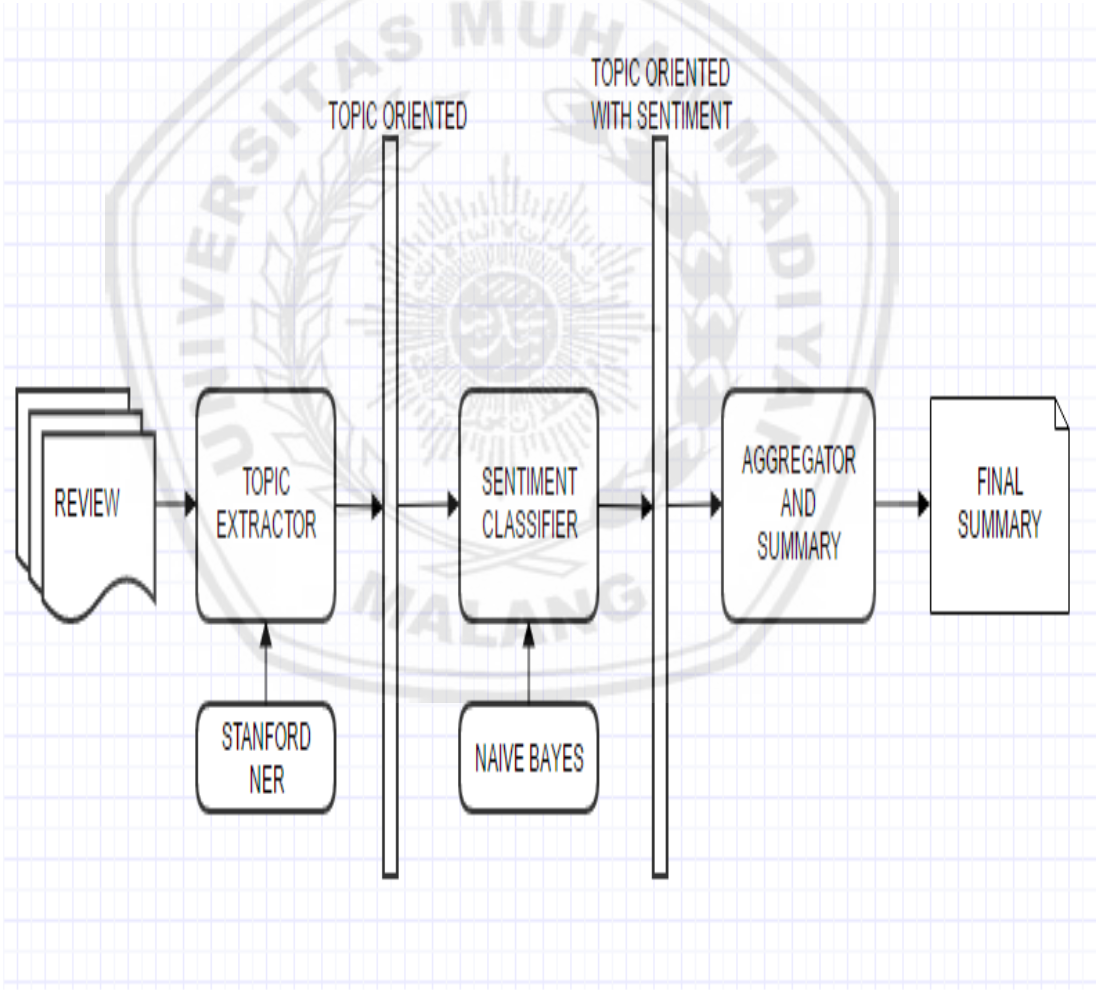
Pada tahap ini, data melewati proses tagging, serta pembuatan model berdasarkan data tagging tersebut, lalu data keseluruhan di pecah berdasarkan aspek yang sedang di bahas, dan selajutnya di lakukan persiapan data test dan data train untuk proses *Analysis Sentiment*, dan proses di atas tersebut di lakukan secara manual. Berikut proses dari hasil analisa pada table di bawah yang menjelaskan pembagian data train dan data test dari data keseluruhan yang telah di pecah menjadi kalimat berdasarkan aspek yang sedang di bahas :

tabel 3 1 Pembagian Data Keseluruhan

Data	Positif	Negatif	Jumlah
Data train	303	106	409
Data test	63	34	92
			500

3.2 Perancangan Sistem

Berikut gambar di bawah merupakan gambaran alur perancangan sistem pada penelitian ini :



gambar 3 1 Alur Perancangan Sistem

Gambar di atas menjelaskan, setelah data di lakukan analisis, selanjutnya data menjadi data yang siap untuk di lakukannya penelitian, data yang siap akan melewati proses

tagset dengan aturan proses *NER* (*Named Entity Recognition*), pada umumnya proses *NER* adalah proses mengekstraksi informasi yang terdapat pada kalimat, bentuk umum atau aturan umum pada *NER* di gunakan untuk mengekstarksi informasi dengan contoh "nama","lokasi","organisasi"[9]. Berikut bentuk aturan umum *NER* pada table di bawah ini :

tabel 3 2 Bentuk Aturan Umum NER

No	Tag	Contoh
1	X	-
2	P = PERSON	Soraya, Ikbal, Madi
3	L = LOCATION	Jakarta, Rumah, Jln Otista
4	O = ORGANIZATION	BNN, BIN, BMKG

Dan bentuk aturan terkait pada penelitian kali ini :

tabel 3 3 Aturan NER Terkait Penelitian

No	Tak	Contoh
1	X	NULL
2	F = FOOD	Ayam, Nasi Goreng, Bakso, sayur dll
3	P = PRICE	40 ribu, 30k, dll
4	S = SERVICE	Pelayanan, antirian, dll

Selanjutnya akan di lakukan pengelompokan berdasarkan aspek menggunakan *naïve bayes classifier*, *Naïve Bayes Classifier* merupakan sebuah metode klasifikasi, ciri utama dari *Naïve Bayes Classifier* ini adalah asumsi yang sangat kuat dari masing - masing kondisi/kejadian. Pada teorema Bayes, bila terdapat dua kejadian yang terpisah (misalkan A dan B), maka teorema Bayes dirumuskan pada persamaan sebagai berikut:

$$P(Ci) = \frac{fD(Ci)}{|D|}$$

Keterangan :

fD(Ci) : Frekuensi dokumen yang memiliki kategori Ci.

|D| : Jumlah seluruh dokumen latih.

$$P(Wkj|Ci) = \frac{f(Wkj|Ci) + 1}{f(Ci) + |W|}$$

Keterangan :

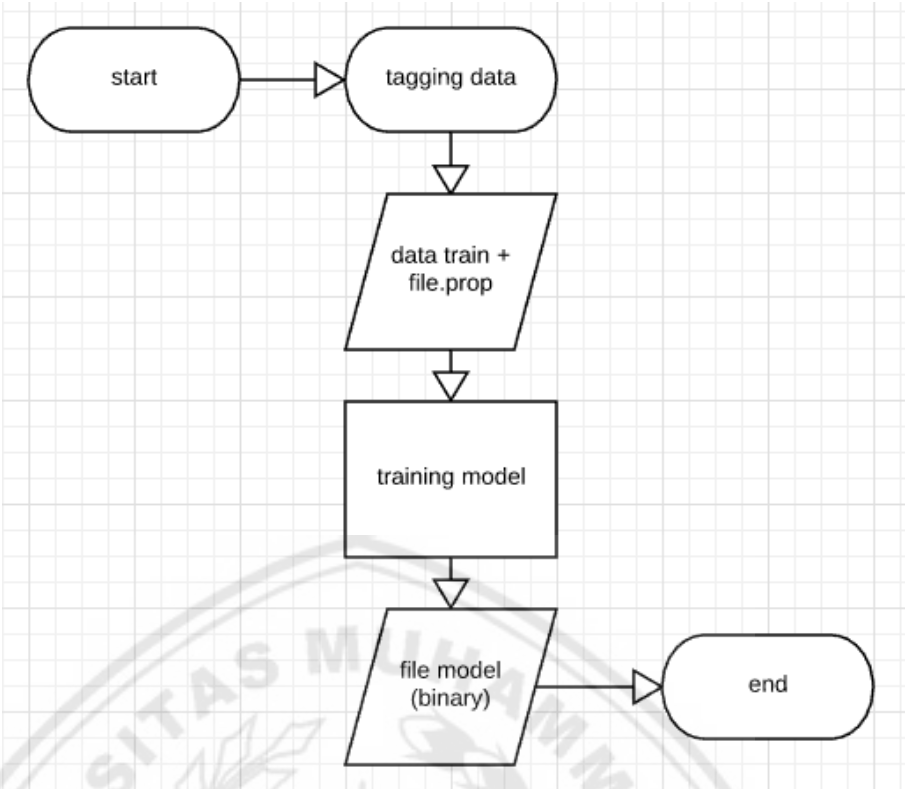
Wkj : frekuensi kata ke-k pada setiap kategori.

W : jumlah kata pada dokumen test.

f(Ci) : frekuensi dokumen berkategori kelas Ci.

3.2.1 Analisa Pembuatan Model NER Berdasarkan Review Restoran

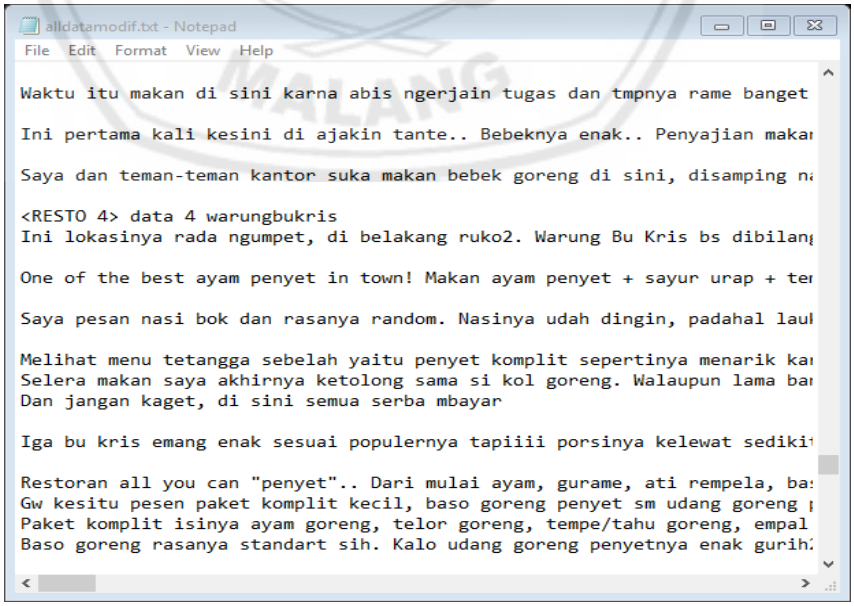
Stanford NER merupakan tools atau aplikasi hasil implementasi java dari *Entifield Entity Recognizer* yang di kembangkan oleh the *Stanford natural language processing group di Stanford university*. *Stanford NER* juga dikenal sebagai *CRFClassifier*. Perangkat lunak ini menyediakan implementasi umum dari urutan acak (*Conditional Random Field* / *CRF*) urutan model. Artinya, dengan melatih model pada data berlabel, menjelaskan bahwa pada penelitian penelitian sebelumnya dapat dengan benar menggunakan kode ini untuk membuat model urutan untuk AP atau penelitian lainnya. Perancangan model terkait penelitian ini dapat di lihat pada gambar di bawah ini :



gambar 3 2 Alur Perancangan Model NER

- Tagging data

Data atau *review* ulasan restoran yang berasal dari zomato.com akan di lakukan tagging dengan cara manual, dengan aturan tagset yang sudah di bahas pada bagian atas seperti berikut. Data ulasan *review* restoran sebelum di lakukan tagging:



gambar 3 3 Data Sebelum Di lakukan Tagging

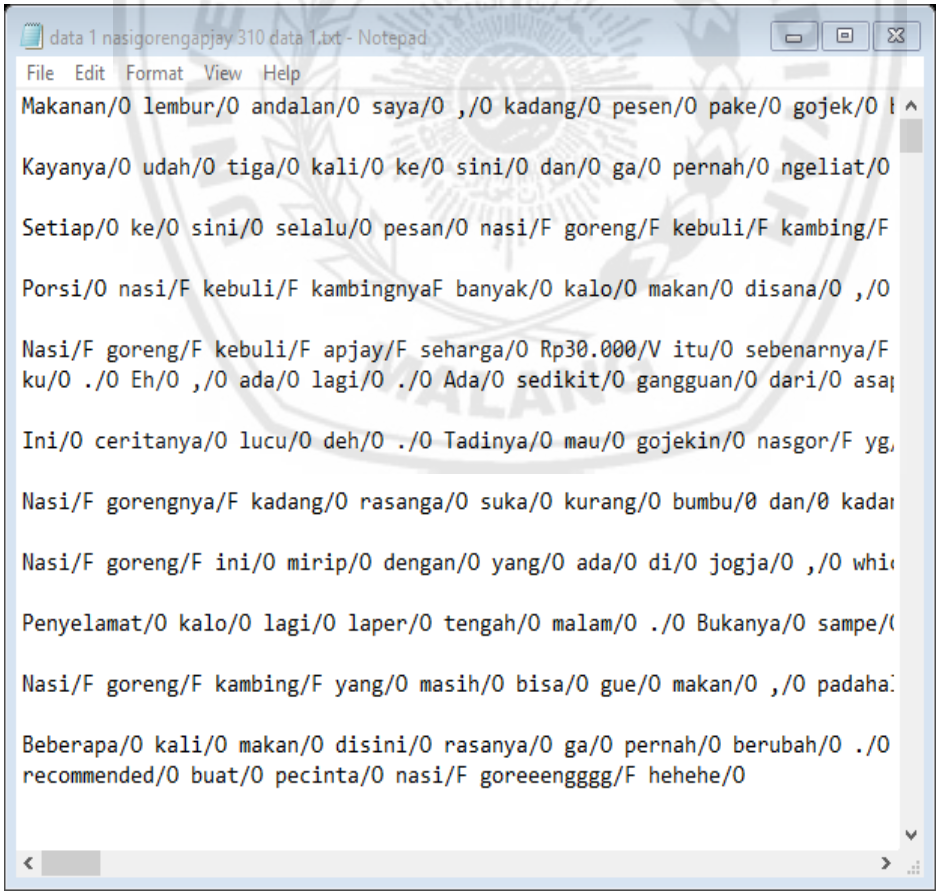
Contoh salah satu dari data di atas sebelum dilakukan tagging.

“ Makanan lembur andalan saya, kadang pesen pake gojek beberapa malam berturut-turut ga bosen-bosen. Agak mahal sih dibanding nasgor lain, tapi bumbunya berasaaaa banget terus dagingnya juga ga pelit! Mienya juga lumayan enak. Tapi tetep favorit nasi gorengnya sih, ga ada yang ngalahin “

Setelah di lakukan tagging maka data akan seperti menjadi di bawah ini.

Makanan/O lembur/O andalan/O saya/O ,/O kadang/O pesen/O pake/O gojek/O beberapa/O malam/O berturut/O -/O turut/O ga/O bosen/O -/O bosen/O ./O Agak/O mahal/O sih/O dibanding/O nasgor/F lain/O ,/O tapi/O bumbunya/O berasaaaa/O banget/O terus/O dagingnya/F juga/O ga/O pelit/O !/O Mienya/F juga/O lumayan/O enak/O ./O Tapi/O tetep/O favorit/O nasi/F gorengnya/F sih/O ,/O ga/O ada/O yang/O ngalahin/O ./O

Dan berikut data yang sudah dilakukan tagging

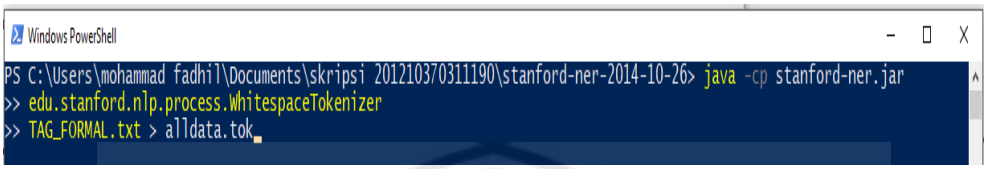


gambar 3 4 Data Setelah DI lakukan Tagging

- Pembuatan file prop

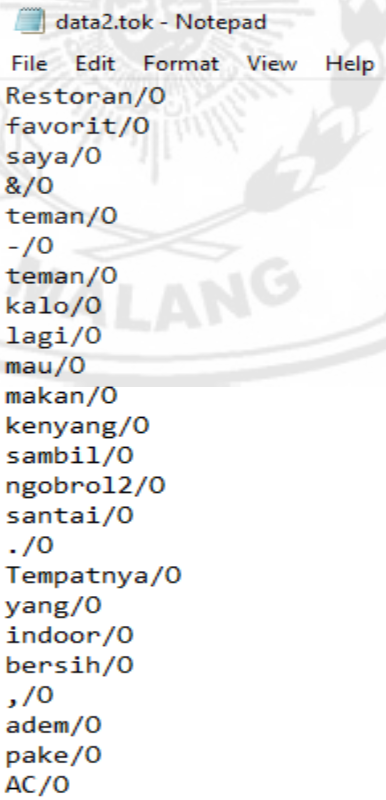
Pada bagian ini data tagging akan melewati proses tokenisasi dan membentuk satu kata per baris dengan menggunakan class Java yaitu *WhitespaceTokenizer*. Class Java tersebut dapat ditemukan pada direktori stanford-ner.jar, dengan cara :

masuk pada file Stanford NER → shift +click kanan → open PowerShell window here → tuliskan perintah berikut .



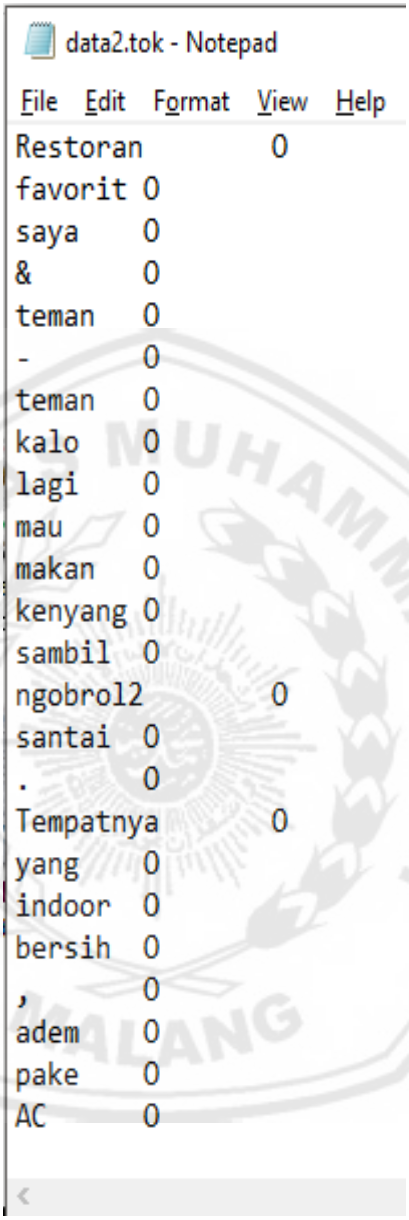
gambar 3 5 Comand Pembuatan File.Tok

Selanjutnya hasil dari proses tersebut akan menghasilkan file.tok, dan file.tok tersebut harus dilakukan penghilangan slash pada token berbaris dengan cara menggunakan aplikasi notepad ++ sehingga file.tok tersebut akan menjadi seperti berikut.



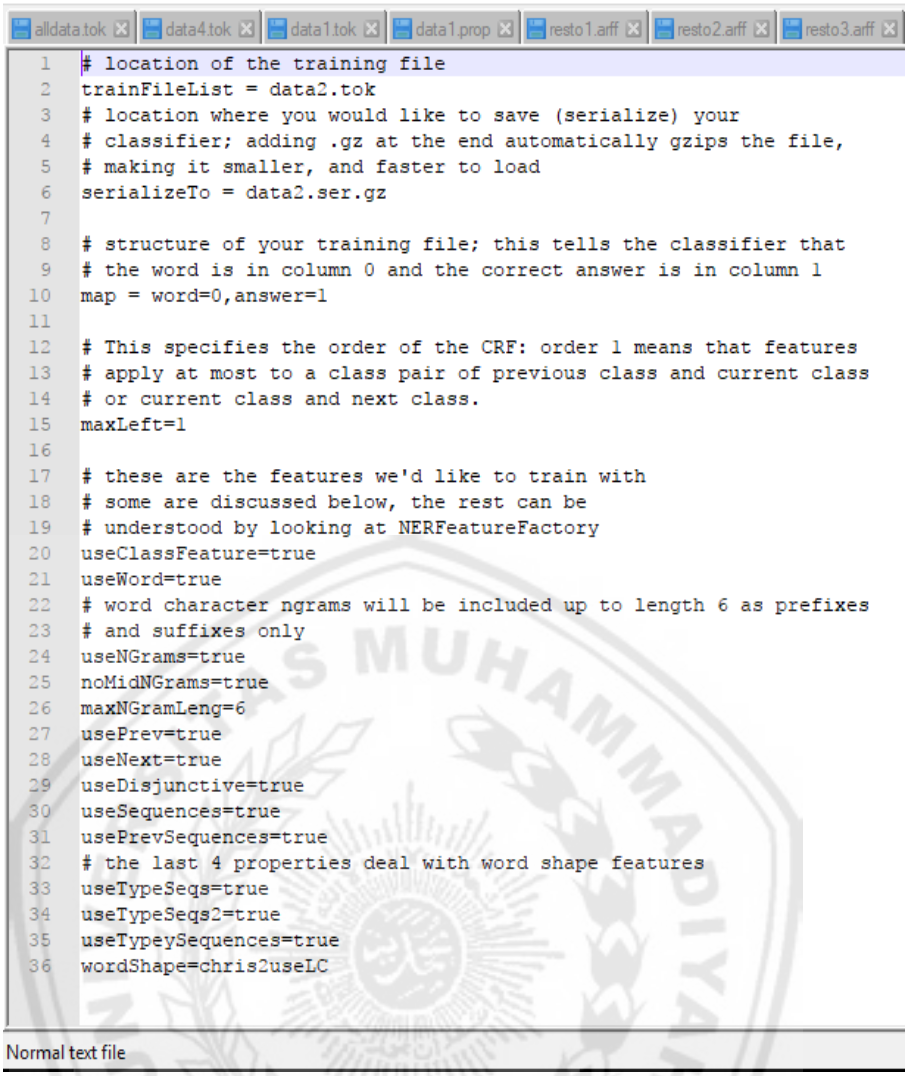
gambar 3 6 Bentuk File.Tok

Slash “ / ” dari file.tok tersebut harus dilakukan penghilangan, dengan menggunakan aplikasi notepad, sehingga file.tok tersebut akan menjadi seperti berikut.



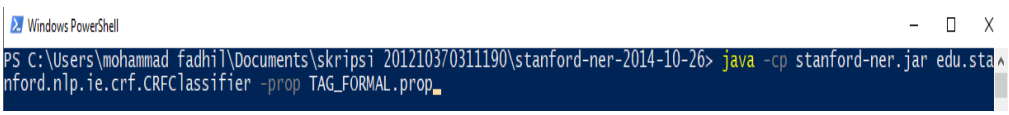
gambar 3 7 Penghilangan Slash File.tok

Selanjutnya proses pembuatan file Prop. File prop adalah Properties File yang berisikan pengaturan file untuk pembuatan model dari data training. Beberapa parameter penting yang akan di-set dalam file ini antara lain trainFileList dan serializeTo. Berikut ini adalah tampilan dari file Props yang telah diubah.



gambar 3 8 Tampilan File.Props yang telah DI ubah

Proses selanjutnya melatih data train dilakukan dengan menggunakan class Java yaitu CRFClassifier. Class Java tersebut dapat ditemukan pada direktori stanford-ner.jar . Berikut ini merupakan command yang akan digunakan:



gambar 3 9 Comand Pembuatan File Props

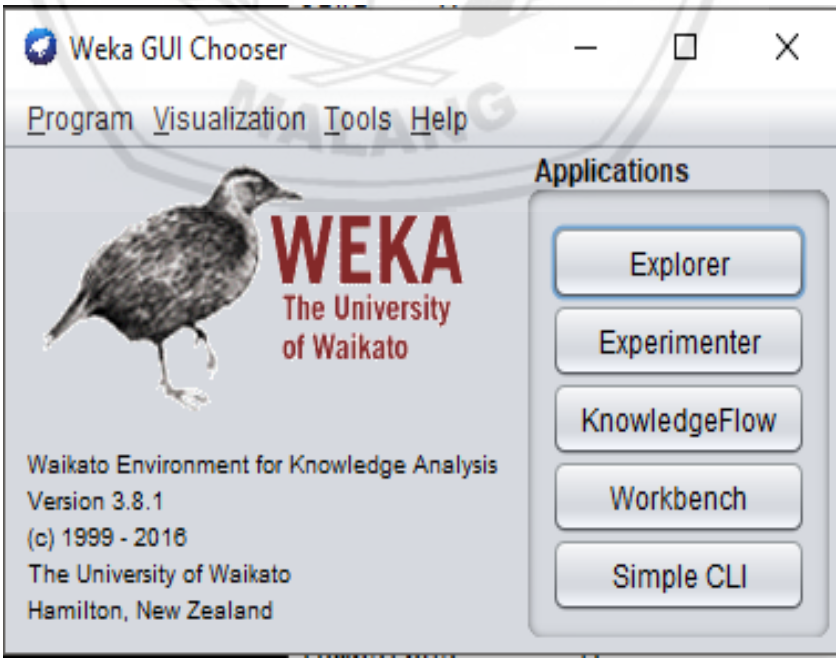
Dan hasil dari proses tersebut akan membentuk model dengan ekstensi .ser.gz. dan file model tersebut berbentuk binner .

3.2.2 Analisa Sentiment Berdasarkan Review Restoran

Pada proses ini, data akan diolah menggunakan weka, weka merupakan aplikasi mining open source berbasis Java. Aplikasi ini dikembangkan pertama kali oleh Universitas Waikato di Selandia baru. Weka terdiri dari koleksi algoritma machine learning yang dapat digunakan untuk melakukan generalisasi / formulasi dari sekumpulan data sampling.

Pada awal proses ini data harus di persiapkan sesuai pembahasan pada bagian 3.1 di atas, sebab kesuksesan penelitian ini terletak pada bagaimana persiapan data yang harus di lakukan untuk menjamin keakuratan formulasi yang di harapkan.

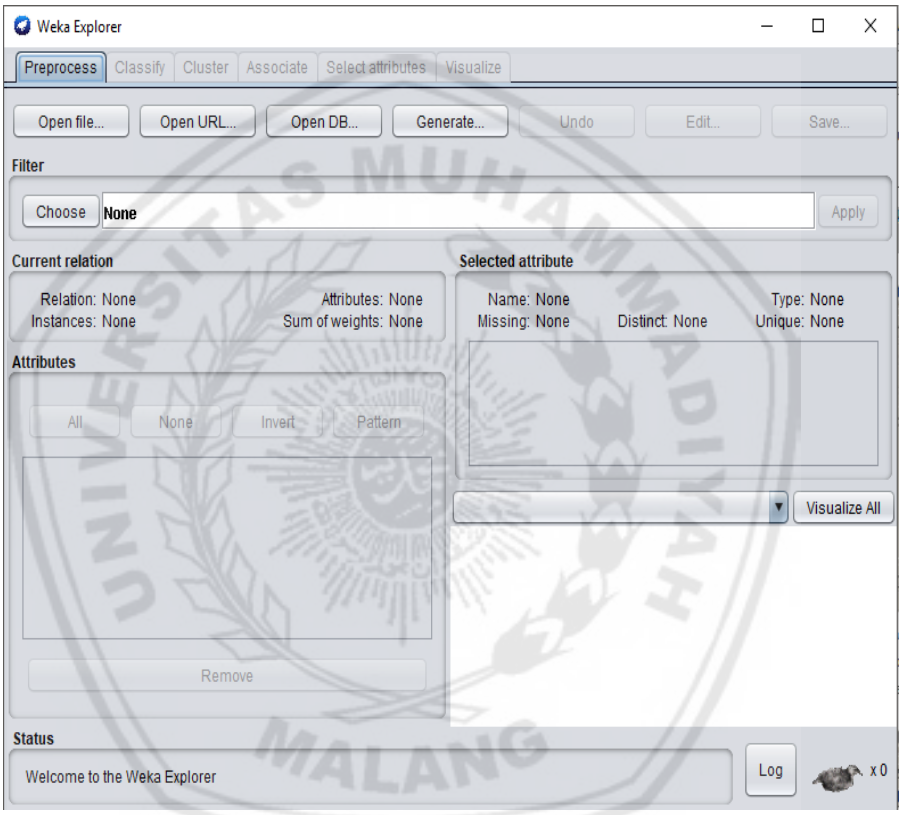
Setelah data siap data akan melewati proses *preprocessing*, *preprocessing* adalah suatu proses pengubahan bentuk data yang belum terstruktur menjadi data yang terstruktur sesuai dengan kebutuhan, untuk proses mining yang lebih lanjut seperti *Analysis Sentiment*, peringkasan, *clustering* dokumen, dll. Konsentrasi penelitian ini merujuk pada *Analysis Sentiment* dimana pengklasifikasian teks berdasarkan sentiment positif dan negative. Berikut *Analysis Sentiment* terkait penelitian ini :



gambar 3 10 Tampilan Awal Weka

Dari gambar di atas terdapat beberapa button untuk menjalankan aplikasi :

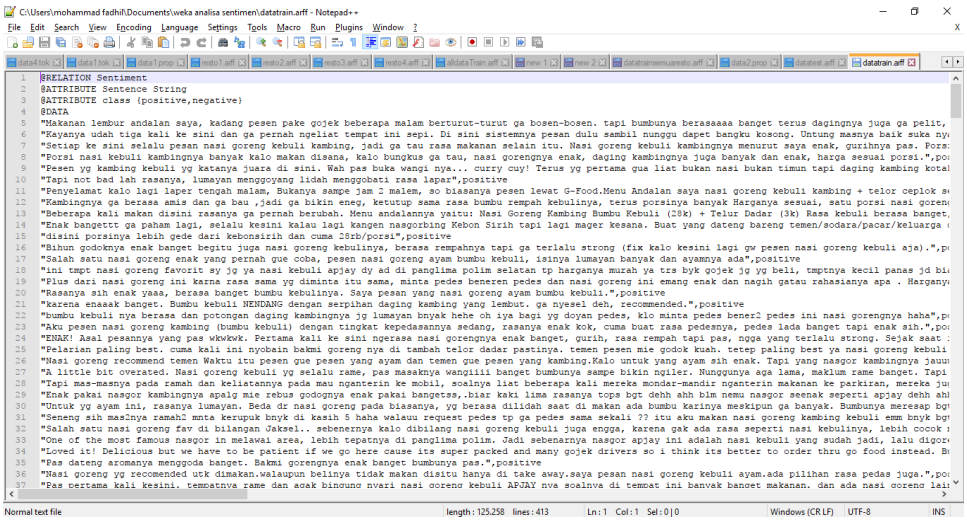
1) *Explorer* digunkan untuk menggali lebih jauh data dengan aplikasi WEKA
2) *Experimenter* digunakan untuk melakukan percobaan dengan pengujian statistic skema belajar
3) *Knowledge Flow* digunakan untuk pengetahuan pendukung
4) *Simple CLI* antar muka dengan menggunakan tampilan *command-line* yang memungkinkan langsung mengeksekusi perintah weka untuk Sistem Operasi yg tidak menyediakan secara langsung



gambar 3 11 Persiapan Preprocess Pada Weka

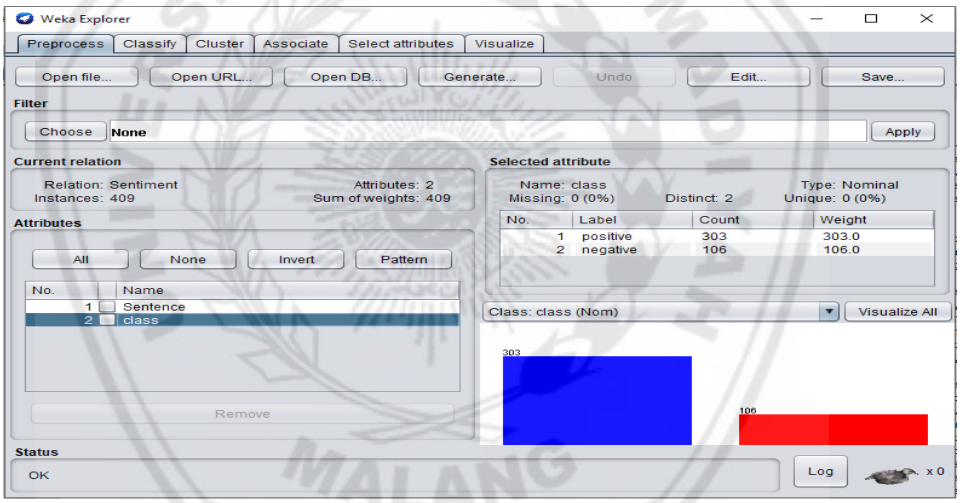
Pada bagian atas window, tepatnya pada bawah judul bar. Terdapat deretan pengolahan data, seperti Prepocess, Classify, Cluster, Associate, Select Attributes Visualize. Namun yang aktif hanya Prepocess ini dikarenakan sebelum menggunakan algoritma diatas pastikan sudah melakukan set file yang akan dieksekusi.

Buka file yang akan dieksekusi oleh Weka, pergunakan extensi file yang di gunakan pada penelitian ini adalah .arff .arff adalah file teks ASCII yang menjelaskan daftar contoh yang berbagi seperangkat atribut. Berikut data train dan data test terkait penelitian ini :



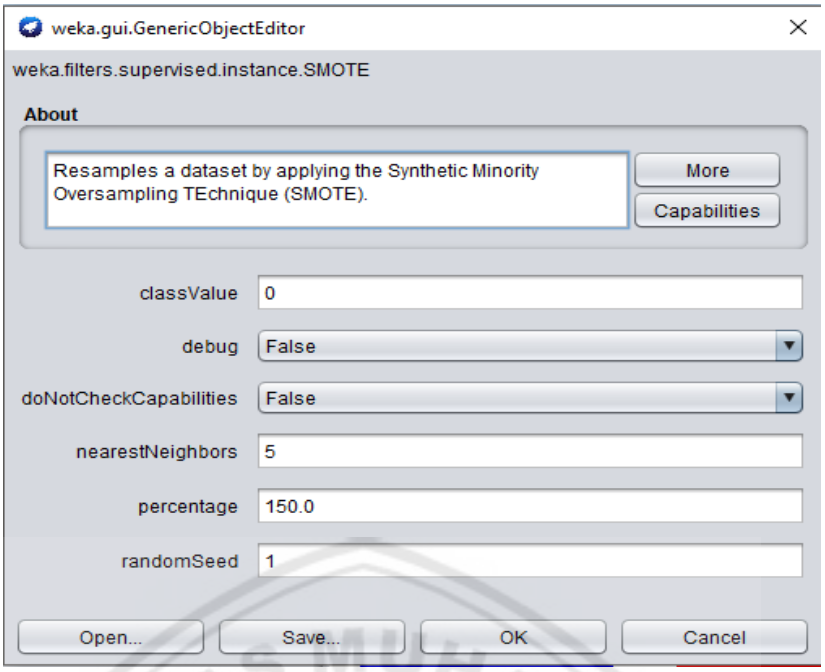
gambar 3 12 Data Training Set

Gambar di atas adalah data training set yang akan melewati preprocessing Pada aplikasi weka seperti berikut ini :



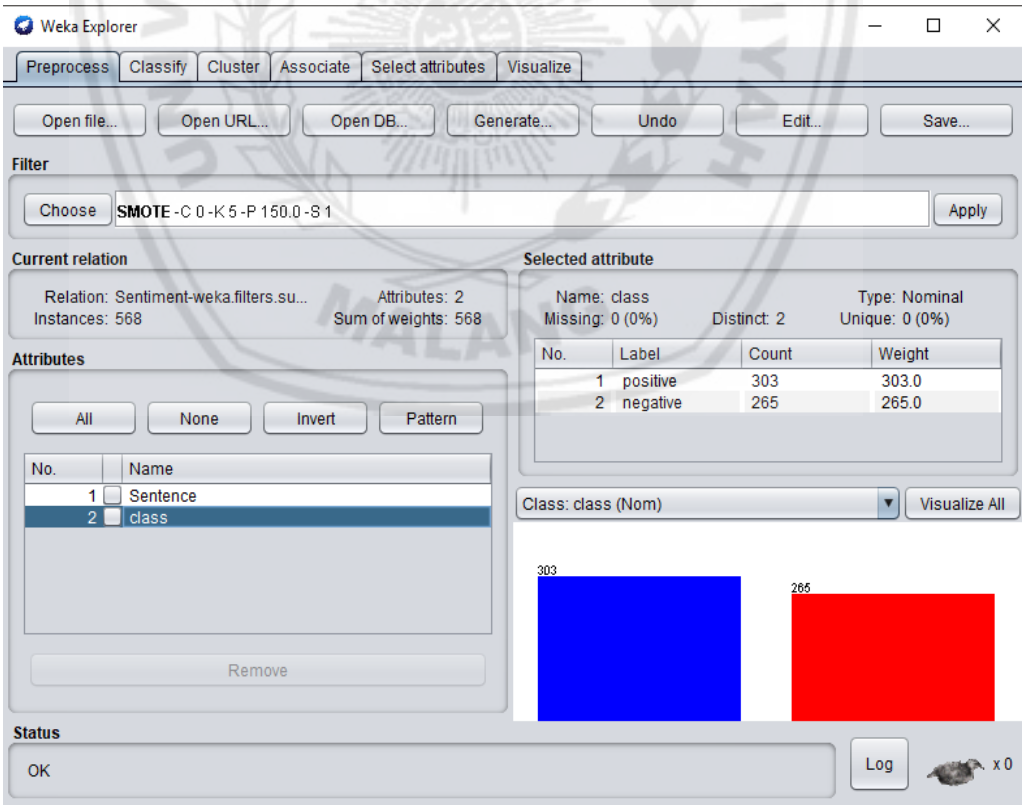
gambar 3 13 Data Training Sebelum Di lakukan Teknik SMOTE

Dari gambar data tersebut, terdapat 2 *class sentiment* yaitu positif dengan 303 data dan negative 106 data, dan dari gambar data tersebut, data mengalami *imbalance class*, *imbalance class* merupakan ketidak seimbangan *class* dalam jumlah data antara dua kelas yang berbeda, salah satu kelasnya merepresentasikan jumlah data yang sangat besar (*majority class*) sedangkan kelas yang lainya merepresentasikan jumlah data yang sangat kecil (*minority class*), dan ketidak seimbangan kelas akan berdampak pada hasil pengujian, sehingga untuk menangani ketidak seimbangan class pada penlitian ini di gunakan teknik SMOTE pada weka sehingga data menjadi balance .



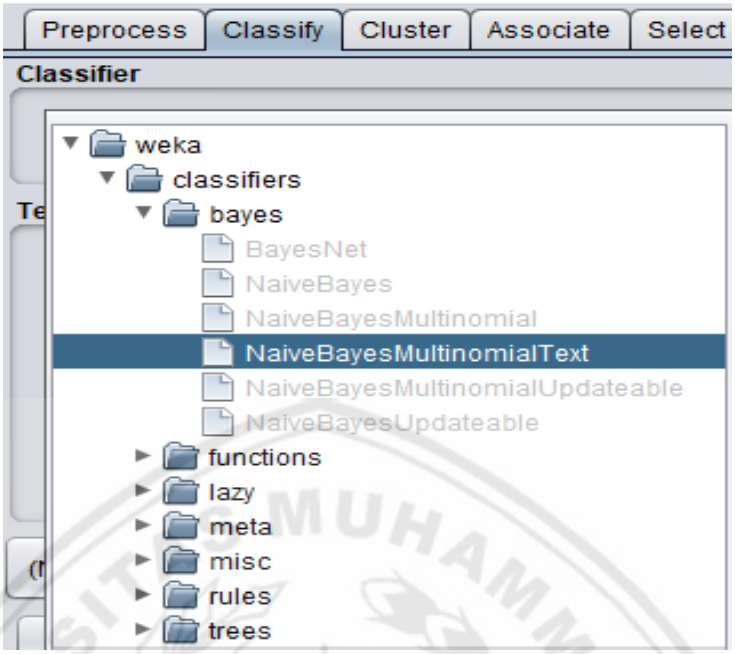
gambar 3 14 Konfigurasi SMOTE

Data di lakukan peningkatan 150.0 pada percentage sehingga sehingga data menjadi *balance* seperti gambar berikut :



gambar 3 15 Data Training Setelah DI lakukan Teknik SMOTE

selanjutnya, metode klasifikasi yang di gunakan pada penelitian ini adalah *naïve bayes* .



gambar 3 16 Pemilihan Metode Klasifikasi

3.3 Analisa Metode Naïve Bayes Model Multinomial Text

Naïve bayes adalah sebuah metode klasifikasi yang dimana prosesnya sederhana dan mudah untuk di pahami begitu juga untuk pengaplikasiannya pada setiap keadaan serta secara umum metode ini dapat mengurangi tingkat kegagalan persentasi yang kecil.

Pada peneltian ini di gunakan pengklasifikasian *naïve bayes* dengan model *multinomial*, pada model *multinomial*, setiap data uji di hitung frekuensi kata pada tiap data latih sentiment c (Tct) dan jumlah kata unik pada data uji (N). Kemudian dari data latih, jumlah kata (∑t'€vTct') dan jumlah *vocabulary* (B) dihitung. *vocabulary* (B) merupakan jumlah kata pada keseluruhan data latih. Setelah frekuensi kata, jumlah kata, dan jumlah vocabulary didapatkan selanjutnya akan dihitung nilai peluang untuk kata masuk pada *sentiment* c(P(t|c)). Perhitungan peluang ini dapat bernilai nol apabila frekuensi katapada data uji di sentimenc (Tct) bernilai nol[10]. Untuk mengatasi masalah ini, dimasukkan koreksi seperti *Laplace smoothing* di semua probabilitas perkiraan sehingga peluang tidak bernilai nol . Perhitungan peluang model *Multinomial* dapat dilihat pada persamaan berikut:

$$P(t|c) = \frac{Tct+1}{(\sum_{t' \in V} Tct')+B}$$

Setelah didapatkan nilai peluang tiap kata, selanjutnya adalah mengalikan nilai peluang tiap katadengan memberikan pangkat (n) sesuai nilai frekuensi kemunculan katatersebut pada data uji. Hasilnya dikalikan kembali dengan nilai peluang tiap sentiment (P(c)). Nilai peluang tiap sentimen (P(c)) dihitung dengan cara membagi jumlah data latih tiap sentimen dengan jumlah data latih keseluruhan sentimen. Perhitungan akhir peluang kalimat pada sebuah sentimen dapat dilihat pada Persamaan 2. *Laplace smoothing* digunakan agar tidak ada kata uji yang berpeluang nol. Hasil yang diambil adalah hasil perkalian yang terbesar pada setiap sentimennya.

$$P(c|d) = \prod_{i=1}^{N} P(t|c)^{n} * P(c)$$

Berikut contoh kasus singkat yang akan di paparkan pada penjelasan analisis *naïve bayes* model *multinomial*

tabel 3 4 Penjelasan SIngkat Naive Bayes Multinomial

	doc ID	Kata dalam document	Class = resto
Training set	1	Ayam nasi ayam	Yes
	2	Ayam ayam tahu	Yes
	3	Ayam tempe	Yes
	4	timun kemangi ayam	No
Test set	5	Ayam ayam ayam timun kemangi	?

Pada contoh table di atas menunjukan bahwa data uji di butuhkan untuk pengklasifikasiannya, di asumsikan dengan p^(c) = ¾ dan p^(c') = ¼ selanjutnya akan mengikuti kondisi probabilitas berikut :

P^(ayam|c) = (5+1) / (8+6) = 6/14 = 3/7

P^(timun|c) = P^(kemangi|c) = (0+1) / (8+6) = 1/4

P^(ayam|c') = (1+1) / (3+6) = 2/9

P^(timun|c') = P^(kemangi|c') = (1+1) / (3+6) = 2/9

Probabilitas di atas penyebutnya adalah (8+6) dan (3+6) karna panjang text c dan text c' masing masing 8 dan 3 dan juga karna jumlah constant adalah 6 maka hasil berikutnya :

P^(c|d5) $\propto \quad 3/4 \cdot (3/7)^3 \cdot 1/14 \cdot 1/14 \approx 0.0003.$

P^(c'|d5) $\propto \quad 1/4 \cdot (2/9)^3 \cdot 2/9 \cdot 2/9 \approx 0.0001.$

Dari hasil tersebut klasifikasi memberikan dokumen uji class = resto, keputusan klasifikasi pada kejadian tersebut terdapat 3 indikator positif pada resto lebih besar dari 2 indikator kejadian negative pada kemangi dan timun.